

Inside the Research C# Compiler

David R. Hanson and Todd A. Proebsting
Programming Language Systems Group
Microsoft Research

<http://www.research.microsoft.com/pls/>

Motivation

- C# presents an opportunity for new language features to see wide use in real applications
- Language design research is experimental—MS programmers don't buy paper designs
- .NET provides much of the required infrastructure
- Missing piece: a C# compiler that is
 - ◆ Complete
 - ◆ Flexible
 - ◆ Easy to understand
 - ◆ Easy to modify
 - ◆ Reasonably efficient

Results in a Nutshell

- Research C# Compiler aka lsc (local Csharp compiler)
- Written in C#; grok in 1-2 weeks
- Stats:
 - 6735 lines lexer, parser, data structures (required)
 - 6946 traditional semantic passes (required for "normal" compiler)
 - 8299 optional utility passes
 - 526 program generation tools
- Simple tools: small 100s of lines
- Complete passes, complex tools: small 1000s of lines
- Typical language extensions: 500-1000 lines; sky's the limit
- Surprise: Most usage to date is for C# tools, not language design

Design From 30,000 Feet

- Dead simple
 - ◆ Parser reads C# source, builds an Abstract Syntax Tree (AST)
 - ◆ Subsequent passes traverse/modify/annotate AST
- Loosely coupled
 - ◆ Passes specified at execution time by command-line options
 - ◆ No fixed passes, no fixed order
 - ◆ No specified output or output format
- Tool-Driven
 - ◆ Parser generator
 - ◆ Generator for pass skeletons
 - ◆ Strongly typed list generator (poor man's generics)

Parsing: Source to AST

- Generalized LR parser generated automatically from possibly ambiguous C# grammar
- Parser
 - ◆ Accepts C# source
 - ◆ Builds parse trees (v1 if ambiguous)
 - ◆ Builds AST from selected parse tree using semantic actions to build AST nodes
- Grammar
 - ◆ Follows published C# grammar closely
 - ◆ Specified in Excel spreadsheet ([Demo](#))
 - ◆ Excel formulas help get types correct
- Hand-written lexer

File Edit View Insert Format Tools Data Window Help

File Edit View Insert Format Tools Data Window Help

File Edit View Insert Format Tools Data Window Help

File Edit View Insert Format Tools Data Window Help

	A	B	C	
185	variable_initializer	array_initializer	variable_initializer	a1
186	local_constant_declaration	const_node constant_declaration	statement	new const_statement(a1)
187	constant_declarations	constant_declaration	List	new const_declaration(a1)
188	constant_declarations	constant_declaration constant_declaration	List	List Cons a1 a2
189	constant_declaration	identifier constant_expression	const_declaration	new const_declaration(a1)
190	expression_statement	statement_expression	statement	new expression_statement(a1)
191	statement_expression	invocation_expression	expression	a1 annotation(a1)
192	statement_expression	object_delegate_creation_expression	expression	a1 annotation(a1)
193	statement_expression	assignment	expression	a1 annotation(a1)
194	statement_expression	post_increment_expression	expression	a1 annotation(a1)
195	statement_expression	post_decrement_expression	expression	a1 annotation(a1)
196	statement_expression	pre_increment_expression	expression	a1 annotation(a1)
197	statement_expression	pre_decrement_expression	expression	a1 annotation(a1)
198	selection_statement	if_statement	statement	
199	selection_statement	switch_statement	statement	
200	if_statement	if (boolean_expression) embedded_statement	statement	new if_statement(a1)
201	if_statement	if (boolean_expression) embedded_statement else	statement	new if_statement(a1)
202	switch_statement	switch (expression) switch_block	statement	new switch_statement(a1)
203	switch_block	{ switch_section }	List	a2
204	switch_section	switch_section	List	switch_section(a1)
205	switch_section	switch_section switch_section	List	List Cons a1 a2
206	switch_section	switch_label statement_list	switch_section	new switch_section(a1)
207	switch_label	switch_label	List	switch_label(a1)
208	switch_label	switch_label switch_label	List	List Cons a1 a2
209	switch_label	case constant_expression	switch_label	new switch_label(a1)
210	switch_label	default	switch_label	new switch_label(a1)
211	throw_statement	throw_statement	statement	

	A	B	C	
185	variable-initializer	array-initializer	variable_initializer	a1
186	local-constant-declaration	constant-local-constant-declaration	statement	new const_statement
187	constant-declaration	constant-declaration	List	declarationListNew
188	constant-declaration	constant-declaration	List	ListConst a1 a2
189	constant-declaration	identifier a constant-expression	const_declaration	new const_declaration
190	statement-expression	statement-expression	statement	new expression_statement
191	statement-expression	invocation-expression	expression	a1 annotation a2
192	statement-expression	object-delegate-creation-expression	expression	a1 annotation a2
193	statement-expression	assignment	expression	a1 annotation a2
194	statement-expression	post-increment-expression	expression	a1 annotation a2
195	statement-expression	post-decrement-expression	expression	a1 annotation a2
196	statement-expression	pre-increment-expression	expression	a1 annotation a2
197	statement-expression	pre-decrement-expression	expression	a1 annotation a2
198	selection-statement	if-statement	statement	
199	selection-statement	switch-statement	statement	
200	if-statement	if (boolean-expression) embedded-statement	statement	new if_statement(a3 a4)
201	if-statement	if (boolean-expression) embedded-statement else	statement	new if_statement(a3 a4 a5)
202	switch-statement	switch (expression) switch-block	statement	new switch_statement(a2)
203	switch-block	{ switch-section ₁ }	List	a2
204	switch-section	switch-section	List	switch_sectionListNew
205	switch-section	switch-section switch-exception	List	ListConst a1 a2
206	switch-section	switch-labels statement-list	switch_section	new switch_section(a3)
207	switch-label	switch-label	List	switch_labelListNew
208	switch-label	switch-labels switch-label	List	ListConst a1 a2
209	switch-label	case constant-expression	switch_label	new switch_expression(a3)
210	switch-label	default	switch_label	new switch_default(a3)
211	when-statement	when-statement	statement	

Microsoft Excel - CSharp-gramm... [Read-Only]				
File Home Layout References Tools Data Window Help				
CSharp-gramm... 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000				
A B C				
150 attribute-name	type-name	type		
151 attribute-arguments	/ positional-argument-list	attribute_arguments	new attribute_arguments	
152 attribute-arguments	/ positional-argument-list named-argument-list	attribute_arguments	new attribute_arguments	
153 attribute-arguments	/ named-argument-list	attribute_arguments	new attribute_arguments	
154 positional-arguments-list	positional-argument	List	expressionList newList	
155 positional-arguments-list	positional-argument-list positional-argument	List	List.Concat a1 a2	
156 positional-argument	attribute-argument-expression	expression		
157 named-argument-list	named-argument	List	named_argumentList	
158 named-argument-list	named-argument-list named-argument	List	List.Concat a1 a2	
159 named-argument	identifier * attribute-argument-expression	named_argument	new named_argument	
160 attribute-argument-expression	conditional-expression	expression		
161 type	pointer-type	type		
162 pointer-type	unmanaged-type *	type	new pointer_type a1	
163 pointer-type	void *	type	new void_pointer_type	
164 unmanaged-type	type	type		
165 primary-expression	pointer-member-access	expression		
166 unary-expression	address-of-expression	expression		
167 pointer-member-access	primary-expression * identifier	expression	new pointer_access	
168 address-of-expression	& unary-expression	expression	new unary_expression	
169 sizeof-expression	sizeof unmanaged-type *	expression	new sizeof_expression	
170 unscoped-statement	fixed-statement	statement		
171 fixed-statement	fixed / pointer-type fixed-pointer-declarator / and statement		new fixed_statement	
172 fixed-pointer-declarator	fixed-pointer-declarator	List	declaratorList newList	
173 fixed-pointer-declarator	fixed-pointer-declarator fixed-pointer-declarator	List	List.Concat a1 a2	
174 fixed-pointer-declarator	identifier * expression	fixed_declarator	new fixed_declarator	
175 variable-initializer	stackalloc-initializer	variable_initializer		
176 stackalloc-initializer	stackalloc unmanaged-type [expression]	variable_initializer	new Stackalloc_initializer	
177				

Parsing: Source to AST

- Generalized LR parser generated automatically from possibly ambiguous C# grammar
- Parser
 - ◆ Accepts C# source
 - ◆ Builds parse trees (+1 if ambiguous)
 - ◆ Builds AST from selected parse tree using semantic actions to build AST nodes
- Grammar
 - ◆ Follows published C# grammar closely
 - ◆ Specified in Excel spreadsheet ([Demo](#))
 - ◆ Excel formulas help get types correct
- Hand-written lexer

Abstract Syntax Trees

- 164 node types; 16 are abstract
- Follows C# syntax closely; strongly typed

```
public class if_statement: statement {  
    public if_statement(expression expr,  
        statement thenpart, statement elsepart) {  
        this.expr = expr;  
        this.thenpart = thenpart;  
        this.elsepart = elsepart;  
    }  
    public expression expr;  
    public statement thenpart;  
    [MayBeNull] public statement elsepart;  
    public override void visit(ASTVisitor prefix,  
        ASTVisitor postfix) { ... }  
}
```

- Lists are strongly typed [Demo]

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809	810	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837	838	839	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864	865	866	867	868	869	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959	960	961	962	963	964	965	966	967	968	969	970	971	972	973	974	975	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991	992	993	994	995	996	997	998	999	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023	1024	1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036	1037	1038	1039	1040	1041	1042	1043	1044	1045	1046	1047	1048	1049	1050	1051	1052	1053	1054	1055	1056	1057	1058	1059	1060	1061	1062	1063	1064	1065	1066	1067	1068	1069	1070	1071	1072	1073	1074	1075	1076	1077	1078	1079	1080	1081	1082	1083	1084	1085	1086	1087	1088	1089	1090	1091	1092	1093	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103	1104	1105	1106	1107	1108	1109	1110	1111	1112	1113	1114	1115	1116	1117	1118	1119	1120	1121	1122	1123	1124	1125	1126	1127	1128	1129	1130	1131	1132	1133	1134	1135	1136	1137	1138	1139	1140	1141	1142	1143	1144	1145	1146	1147	1148	1149	1150	1151	1152	1153	1154	1155	1156	1157	1158	1159	1160	1161	1162	1163	1164	1165	1166	1167	1168	1169	1170	1171	1172	1173	1174	1175	1176	1177	1178	1179	1180	1181	1182	1183	1184	1185	1186	1187	1188	1189	1190	1191	1192	1193	1194	1195	1196	1197	1198	1199	1200	1201	1202	1203	1204	1205	1206	1207	1208	1209	1210	1211	1212	1213	1214	1215	1216	1217	1218	1219	1220	1221	1222	1223	1224	1225	1226	1227	1228	1229	1230	1231	1232	1233	1234	1235	1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247	1248	1249	1250	1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263	1264	1265	1266	1267	1268	1269	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	1290	1291	1292	1293	1294	1295	1296	1297	1298	1299	1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	1310	1311	1312	1313	1314	1315	1316	1317	1318	1319	1320	1321	1322	1323	1324	1325	1326	1327	1328	1329	1330	1331	1332	1333	1334	1335	1336	1337	1338	1339	1340	1341	1342	1343	1344	1345	1346	1347	1348	1349	1350	1351	1352	1353	1354	1355	1356	1357	1358	1359	1360	1361	1362	1363	1364	1365	1366	1367	1368	1369	1370	1371	1372	1373	1374	1375	1376	1377	1378	1379	1380	1381	1382	1383	1384	1385	1386	1387	1388	1389	1390	1391	1392	1393	1394	1395	1396	1397	1398	1399	1400	1401	1402	1403	1404	1405	1406	1407	1408	1409	1410	1411	1412	1413	1414	1415	1416	1417	1418	1419	1420	1421	1422	1423	1424	1425	1426	1427	1428	1429	1430	1431	1432	1433	1434	1435	1436	1437	1438	1439	1440	1441	1442	1443	1444	1445	1446	1447	1448	1449	1450	1451	1452	1453	1454	1455	1456	1457	1458	1459	1460	1461	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	1482	1483	1484	1485	1486	1487	1488	1489	1490	1491
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------

Abstract Syntax Trees

```
public class if_statement statement {
    public if_statement(expression expr
        statement thenpart statement elsepart) {
        this.expr = expr;
        this.thenpart = thenpart;
        this.elsepart = elsepart;
    }
    public expression expr;
    public statement thenpart;
    [Nullable] public statement elsepart;
    public override void visit(ASTVisitor prefix
        ASTVisitor postfix) { }
}
```

Semantics

```
public class if_statement statement {  
    public override void visit(ASTVisitor prefix  
        ASTVisitor postfix) {  
        prefix(this)  
        expr visit(prefix postfix)  
        thenpart visit(prefix postfix)  
        if (elsepart != null)  
            elsepart visit(prefix postfix)  
        postfix(this);  
    }  
}
```

Simple C# Tools

```
public static void prefix() { }
public static void postfix() { }
public static void Main(string[] args) {
    foreach (string s in args) {
        AST ast = (AST)Parser.parse(s,
                                    new StreamReader(s));
        if (ast != null)
            ast.visit(new ASTVisitor(prefix),
                     new ASTVisitor(postfix));
    }
}
```

Finding Assertions

```
if (..) throw
```

```
public static void doit(AST ast) {  
    if (ast is if_statement  
        && ((if_statement)ast).thenpart is throw_statement  
        && ((if_statement)ast).elsepart == null)  
        Console.WriteLine("{0} Possible assertion",  
            ast.begin);  
}
```

```
public static void Main(string[] args) {  
    ast.visit(new ASTVisitor(doit))  
}
```

```
} else
```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1

```

process) [Process] End *

```



```
(process) [Process] End *
```

```
(process) [Process] End *
```

Finding Assertions

```
if (..) throw
```

```
public static void doit(AST ast) {  
    if (ast is if_statement  
        && ((if_statement)ast).thenpart is throw_statement  
        && ((if_statement)ast).elsepart == null)  
        Console.WriteLine( "{0} Possible assertion",  
                             ast.begin);  
}  
  
public static void Main(string[] args) {  
    ast.visit(new ASTVisitor(doit)) ;  
}
```

Leme

Visitors

```
public static object visit(object ast, TextWriter w)
```

-
- ◆ Print output if any to w

```
lsc -visitor
```

-
- ◆
- ◆ mkvisitor -class bind -args SymbolTable bindings >foo.cs
- ◆ Demo

public declaration; let body;

1. The first part of the document discusses the importance of maintaining accurate records of all transactions. It emphasizes that proper record-keeping is essential for the integrity of the financial system and for the ability to detect and prevent fraud.

2. The second part of the document outlines the specific requirements for record-keeping. It states that all transactions must be recorded in a timely and accurate manner, and that the records must be maintained for a minimum of five years.

3. The third part of the document discusses the role of the auditor in verifying the accuracy of the records. It states that the auditor must perform a thorough review of the records to ensure that they are complete and accurate, and that any discrepancies must be reported to the appropriate authorities.

4. The fourth part of the document discusses the consequences of failing to comply with the record-keeping requirements. It states that any individual or entity that fails to maintain accurate records may be subject to fines, penalties, and even criminal prosecution.

5. The fifth part of the document discusses the importance of training and education for individuals involved in record-keeping. It states that all individuals who are responsible for maintaining records must receive appropriate training and education to ensure that they are able to perform their duties accurately and efficiently.

6. The sixth part of the document discusses the importance of regular audits and reviews of the record-keeping system. It states that regular audits and reviews are necessary to ensure that the system is working properly and to identify any areas for improvement.

Page 27

Page 28

SymbolTable binding:

1. The first part of the document discusses the importance of maintaining accurate records of all transactions and activities. It emphasizes that this is crucial for ensuring transparency and accountability in the organization's operations.

2. The second part of the document outlines the various methods and tools used to collect and analyze data. It highlights the need for consistent and reliable data collection processes to support informed decision-making.

3. The third part of the document describes the results of the data analysis and the insights gained from the study. It provides a detailed overview of the findings and their implications for the organization's future strategy.

4. The fourth part of the document discusses the challenges encountered during the research process and the steps taken to overcome them. It provides valuable lessons learned that can be applied to future research projects.

5. The fifth part of the document concludes the report by summarizing the key findings and providing recommendations for further research and action. It emphasizes the importance of ongoing monitoring and evaluation to ensure the organization's continued success.

6. The sixth part of the document provides a detailed appendix of the data collected and the analysis performed. This section is intended to provide a comprehensive overview of the research data and the methods used to analyze it.

7. The seventh part of the document provides a detailed appendix of the literature reviewed and the sources used. This section is intended to provide a comprehensive overview of the research literature and the sources used to inform the study.

8. The eighth part of the document provides a detailed appendix of the research instruments used, including questionnaires, interviews, and focus groups. This section is intended to provide a comprehensive overview of the research instruments used to collect data.

9. The ninth part of the document provides a detailed appendix of the research results and the analysis performed. This section is intended to provide a comprehensive overview of the research results and the analysis performed.

10. The tenth part of the document provides a detailed appendix of the research conclusions and the recommendations made. This section is intended to provide a comprehensive overview of the research conclusions and the recommendations made.

Visitors

```
public static object visit(object ast, TextWriter w)
```

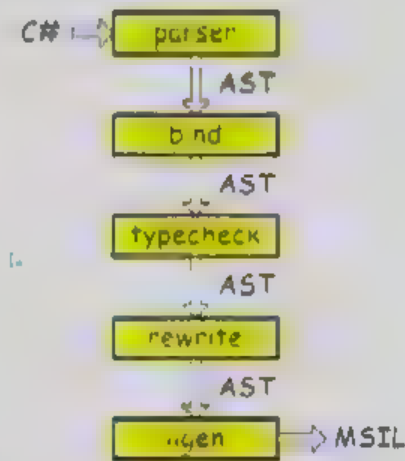
- ◆
- ◆ Print output if any to w

```
lsc -visitor
```

- ◆
- ◆
- ◆ `mkvisitor -class bind -args SymbolTable bindings >foo.cs`
- ◆ `.Demo.`

Traditional Compiler Passes

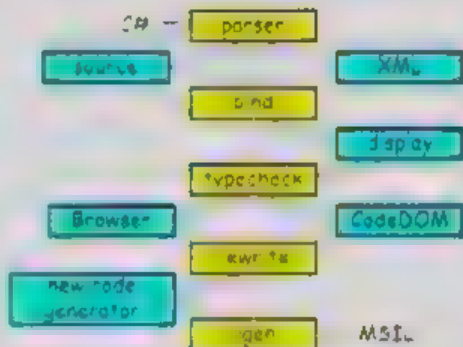
- parse
 annotate with Symbols
- typecheck
 •
 Method s
- rewrite
 •
 generation
- codegen
 emit MSIL



"Exotic" Visitors

different kinds of processing
needed at each

- visitation order



Visitor Examples

```
lsc -visitor XML hello.cs
```

emo

```
lsc -visitor source hello.cs
```

Lemo

```
lsc -visitor sortmembers -visitor source mkvisitor.cs
```

Lemo

```
lsc -visitor bind -visitor display debug/base dli hello.cs
```

emo


```

./find-assert x x grep x tabler x
/mkvisitor -class bind -args SymbolTab x bindings >foo.o
/lose -visitor XML hello.cs
x users dsh csharp compiler inside sh (Shell Fill) Top
<tag><identifier< tag>
    </inputElement>
    </id>
    <sym/>
    <mods>
    * mods*
    <end>hello.cs (0 2)</end>
</class_declaration>
* declarations*
<using_directives>
</using_directives>
<sym/>
    <end>hello.cs (0 2)</end>
</compilation_unit>
</inputs>
<assemblyRefs.*
</compilation>
<end>(0 0)</end>
</program>
[XML: 0 0200s]
[Total: 0 0510s]
X users dsh csharp compiler 2
(process (Process End *

```

File Edit View Window Help

```

mkvisitor -class bind -args 1,mt 1,table bindings >foo.cs
/loco -visitor XMT hello.cs
/loco visitor source hello.cs
* users dir csharp compiler inside sh ,Shell File, 20*
</using_directives>
<sym/>
<end-hello cs(0 2)> end>
</compilation_unit>
</inputs>
<assemblyRefs/>
</compilation>
<end>(0.0)</end>

\ program\
XMT- 0 0200s\
[Total 0 0010s]
\-----\
\loco -visitor source hello.cs
[parse 0 0109s]

class Hello {
    public static void Main()
        System.Console.WriteLine( Hello world )
    |
}

[source: 0 0300s]
[Total: 0 0010s]
* users dir csharp compiler 2
process [Process, End *
```


Visitor Examples

```
lsc -visitor XML hello.cs
```

emo

```
lsc -visitor source hello.cs
```

Lemo

```
lsc -visitor sortmembers -visitor source mkvisitor.cs
```

Lemo

```
lsc -visitor bind -visitor display debug/base dli hello.cs
```

emo

3/7/2003 11:25:52 AM

compilation#7

org\$toStringList(-visitor b nd -visitor d\$ep by debug base dll helio cs)

inputs=

assemblyRef=at ingList()

parent=

compilation_unit#10

compilation_unit#10

compilation_unit#13

attributes=attribute_sectionList#15 empty,

declaration_sectionList#17

using_directives=using_directiveList#19 empty)

symbolNameSpace#

parents=

declarationList#17

class_declaration#23

NameSpace#21

aliases=stringList()

using_namespaceList#

```

class declaration#23
  attribute_section_list#33 (empty)
  bases=TypeList#15 (empty)
  body=declaration#37
  id=InputElement (code object 37 str="Hello my identifier")
  name="pe" #41
  module=InputElement#45 (empty)
  parents=ImportSection#41

```

SymbolTable#27

```

Owner: Module#2
[System's NameSpace#40
{__builtins__}=Module#2
{__doc__}=None
{__name__}='__main__'
{__package__}=None

```

declaration#37

```

method_declaration#51
  name="method" #51
  parameters=List#52
  body=List#53

```

ClassType#43

```

base_classes=List#44
interfaces=InterfaceTypeList#57 (empty)

```


Method#79

```

interfaceMethod#null
local:=SymbolLabel#127 (empty,
signature:=> j / 1 2 3 4 5 6 7
module:=null
attributes:=null
dec Spacer _ sym = " " #70
id:=InputElement(coord=hello cs,4 1) str:=Main tag:=dent + en)
mod f ens:=string_ st(publ : static)
serial:=61

```

InputElement, #781

```

InputElement(coord=hello cs,4 2) str:=pub c tag:=publ c)
InputElement(coord=hello cs,4 9) str:=static tag:=static)

```

block_statement#86

```

stmt:=statementList#136 (empty
symsh _ x = "
method=Method#20
lab=0
parents _ c _ 1 2 3 4 5 6 7 8 9

```

constructor_declaration#88

```

class_declaration#23
  attribute_section_list#33 empty,
  bases=type_list#35 empty,
  body=statement_list#37
  is=InputElement(declaration_list(37) str="Hello, world!",
  sym=class_type#4
  method_declaration_list#43 empty)
  parents=import_list#45

```

Symbol Table#27

```

  Owners: dict#3
  {System}=NameSpace#40
  {file}=class_type#41
  {Microsoft}=class_type#42

```

declaration_list#37

```

  method_declaration#51
  class_declaration#52
  class_declaration#53

```

ClassType#43

```

  base_class=class_type#44
  interfaces=InterfaceTypeList#46 empty

```

Navigating C# Source and ASTs

Some

-
-

```
lsc -visitor Browser BrowserVisitor debug/Browser.dll hello.cs
```

and

```
lsc -visitor bind -visitor typecheck  
-visitor Browser BrowserVisitor debug/Browser.dll Bq.cs
```

Back Forward Parent Add Object Browser Add CodeDom Browser Add L Browser

Sid: hello.cs#1 2002-03-08 11:40:12 REDMOND .NET compiler unit compilation unit 176

class hello {

public static void Main() {

System.Console.WriteLine("hello, world!");

hello.cs

Back Forward Parent Add Object Browser Add CodeDom Browser Add ... Browser

Sid: eno-usa1 2002-03-08 11:40:12 REDMOND .NET

Compilation Unit: compilation-unit#176

class Hello {

public static void Main()

{ System.Console.WriteLine("Hello, world!");

... for all ... section ... empty

... section ... section ...

... directives ... directives ...

...

parent compilation#190

get_end <property>

get_begin <property>

Back	Forward	Parent	Add Object Browser	Add CodeDom Browser	Add L Browser
3id hello.cs#1	2002-03-08 14:40:12	PEDMOND	1 r	compilation unit compilation unit#176	attributes attribute section sim#19 empty
class /hello {				declaration declaration unit#187	using directives using directive sim#185 en
public static void Main() {				sym null	parent compilation#190
System.Console.WriteLine("hello, world");				get_end <property>	get_begin <property>
				declaration list declaration list#182	get_Count <property>
				get_inner list <property>	get_list <property>
				AddDecl & list#183	
				(class declaration) class declaration#1230	

Back	Forward	Parent	Add Object Browser	Add CodeDom Browser	Add Browser
			Sid:Hello.cs#1 2002-03-08 11:40:42 REDMOND JIR	compilation-unit-compilation-unit#176	
				attributes attribute-section-list#179 (empty)	
				declarations declaration-list#182	
				using-directives using-directive-list#185 (empty)	
				parent-compilation#190	
				get_end <property>	
				get_begin <property>	
				declaration-list-declaration-list#182	
				get_Count <property>	
				get_innerList <property>	
				get_list <property>	
				using_directive class-declaration#230	
				using_directive class-declaration#230	
				(string_literal) string_literal#272	
				token-inputElement-coord#hello-05-8-28-string	
				valueUsed True	
				typ null	
				value nul	
				parent-arguments#283	
				get_end <property>	
				get_begin <property>	

Back	Forward	Parent	Add Object Browser	Add CodeDOM Browser	Add XML Browser
Sid:hello.cs#1 2002-03-08 11:40:42 REDMOND .NET			compilation unit compilation unit#176		
class Hello {			attributes attribute section: sim179 empty		
public static void Main ()			declarations declaration unit#182		
System.Console.WriteLine ("hello, world")			using directives using directive: sim185 empty		
}			parent compilation#190		
			get_end <property>		
			get_begin <property>		
			declaration:st declaration: st#182		
			get_Count <property>		
			get_InnerList <property>		
			get_List <property>		
			... class declaration#230		
			class Declaration class declaration#230		
			using namespace using namespace#231		
			token: namespace: 1000: the 0 0 5 3 28 str		
			value: used True		
			type: nu		
			value: nu		
			parent arguments#281		
			get_end <property>		
			get_begin <property>		

Back Forward Parent

Add Object
Browser

Add CodeDom
Browser

Add L
Browser

Sid: hello.cs#1 2002-03-08 11:40:12 REDMOND J...

class Hello {

public static void Main ()

System.Console.WriteLine("hello, world")

declarations declaration_stmt#182

using directives using_directive#185

sym null

parent compilation#190

get_end <property>

get_begin <property>

declaration_list declaration_list#182

get_Count <property>

get_InnerList <property>

get_List <property>

2 class declaration#230

case declaration case declaration#230

string literal string_literal#272

token InputElement coord=hello.cs 528 str

rawUsed True

typ null

value null

parent arguments#289

get_end <property>

get_begin <property>

argument argument#283

value expression value ref expression

hello.cs

Navigating C# Source and ASTs

Example

-
-

```
lsc -visitor Browser BrowserVisitor debug/Browser.dll hello.cs
```

```
... ..
```

```
lsc -visitor bind -visitor typecheck
```

```
  -visitor Browser BrowserVisitor debug/Browser.dll Bq.cs
```

```
... ..
```


Back Forward Parent Add Object Browser Add CodeDom Browser Add L Browser

```
up[i] = down[i] = true
for (int i = 0; i < rows.Length; i++)
    rows[i] = true
queens = new int[8];
```

```
static void Queens(int n) {
    for (int i = 0; i < rows.Length; i++)
        if (rows[i] && up[i] && down[i])
            rows[i] = up[i] = down[i] = false;
    x[c] = i;
    if (c == 7)
        print(x);
    else
        queens[c] = i;
    rows[c] = up[c] = down[c] = true;
}
```

```
static void print(int[] x) {
    foreach (int c in x)
        System.Console.Write("{0} ", c + 1);
    System.Console.WriteLine();
}
```

```
S11
for
maxstack 12
end of method EightQueens queens
method <debug> static private void print
locals init ([0])int32 c
    Bq cs 23 21) x
    locals init ([1])int32 i11
    ccs = m1_2, m32, 12
    idarg 0 x
    stloc 2 i2
    ccs = 0
    stloc 1 i1
    br S20
S17
    Bq cs 23 3) 0 in x
    dloc 2 i2
    dloc 1 i1
    ccs = 4
    stloc 0 c
    ccs = 4 System.Console.WriteLine("{0}"
    dloc 0 c
```


Back Forward Parent Add Object Browser Add CodeDom Browser Add L

```
up[i] = down[i] = true;
for (int i = 0; i < rows.Length; i++)
    rows[i] = true;
queens = new int[8];
```

```
static void Queens(int n) {
    for (int i = 0; i < rows.Length; i++)
        if (rows[i] && up[i] && down[i])
            rows[i] = up[i] = down[i] = false;
        else
            if (i == 0)
                print(x);
            queens = 1;
            rows[i] = up[i] = down[i] = true;
}
```

```
static void print(int[] x) {
    foreach (int c in x)
        System.Console.Write("{0} ", c);
    System.Console.WriteLine();
}
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace EightQueens {
    class Program {
        static void Main() {
            Queens(8);
        }

        static void Queens(int n) {
            int[] rows = new int[n];
            int[] up = new int[n];
            int[] down = new int[n];
            int[] queens = new int[n];
            for (int i = 0; i < rows.Length; i++)
                rows[i] = up[i] = down[i] = false;
            Queens(0, rows, up, down, queens);
        }

        static void Queens(int i, int[] rows, int[] up, int[] down, int[] queens) {
            if (i == rows.Length)
                Print(queens);
            else
                for (int c = 0; c < rows.Length; c++)
                    if (!rows[c] && !up[i + c] && !down[i - c]) {
                        rows[c] = up[i + c] = down[i - c] = true;
                        queens[i] = c;
                        Queens(i + 1, rows, up, down, queens);
                    }
            rows[i] = up[i] = down[i] = false;
        }

        static void Print(int[] queens) {
            Console.WriteLine("Queens: {0}", string.Join(" ", queens));
        }
    }
}
```


Extending C# Adding typeswitch

```
typeswitch (o) {  
    case Int32 x: Console.WriteLine(x). break;  
    case Symbol s: Symbols.Add(s). break;  
    case Segment: popSegment(). break;  
    default: throw new ArgumentException();  
}
```

down to C#

```
if (o is Int32) {  
    int x = (int)o; Console.WriteLine(x);  
} else if (o is Symbol) {  
    Symbol s = (Symbol)o; Symbols.Add(s);  
} else if (o is Segment) { popSegment(); }  
else { throw new InvalidArgument(); }
```

The Easy Way Source-to-Source Transformation

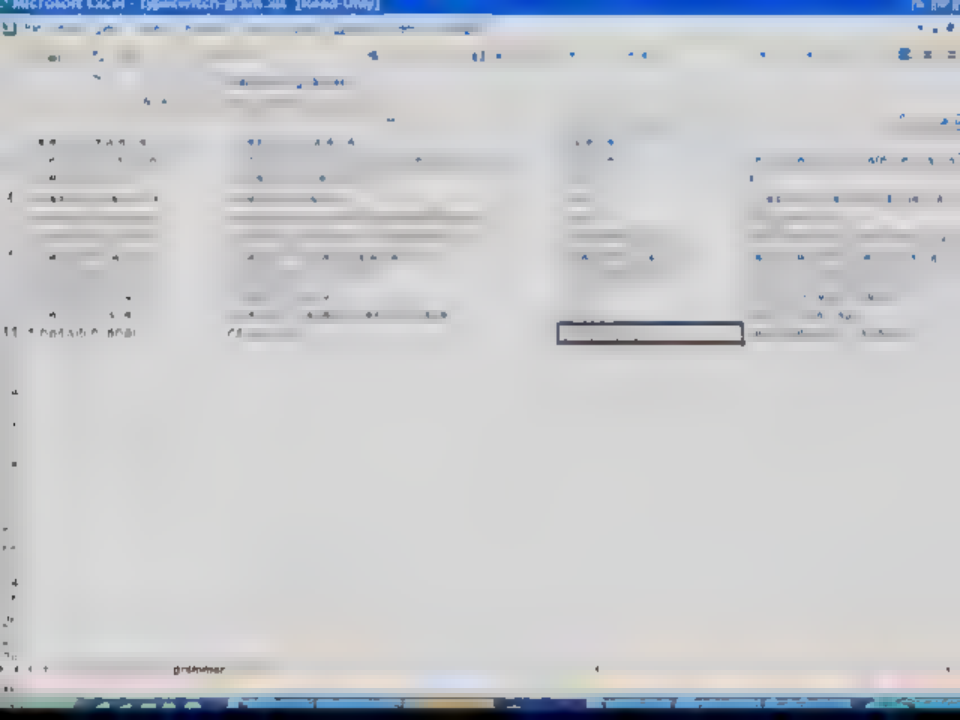
FT

123

2000

A. J. Auerbach

-



1. The first step is to identify the problem.
2. The second step is to define the problem.
3. The third step is to analyze the problem.
4. The fourth step is to develop a solution.

1. The first step is to identify the problem.
2. The second step is to define the problem.
3. The third step is to analyze the problem.
4. The fourth step is to develop a solution.


```

<?xml version="1.0" encoding="utf-8" ?>
<load -visitor XML hello.cs
<load -visitor source hello.cs
<load -visitor sortmembers visitor source mkvisitor.cs >bas.cs
x:\users\dm\sharp\compiler\inside\sh\Shell File, 25>
  </using_directives>
  <sym/>
  <end-hello.cs(0 2)</end>
</compilation_unit>
</inputs>
<assemblyRefs/>
</compilation>
<end>(0.0)</end>
</program>
XML: 0.0200s,
[Total: 0.6810s]
[parse: 0.6109s]

```

```

class Hello {
    public static void Main() {
        System.Console.WriteLine("Hello world")
    }
}
[source: 0.0300s]
[Total: 0.6810s]
M: users\dm\sharp\compiler\inside\sh\Shell File, 25>
(process: [Process, End]

```



```

> [U] [C] [F] [E] [T]
/loac -visitor Browser BrowserVisitor debug Browser dll hello.cs
/loac -visitor kind -visitor typecheck -visitor Browser BrowserVisitor debu
loac -visitor typeswitch source demos typeswitch sample.cs
* users dir sharp compiler inside sh .Shell file, cs-
  < using_directives>
  < sym/>
  < end_hello.cs(0 2)</end>
  < compilation_unit>
  </inputs>
  < assemblyRefs/>
</compilation>
<end>(0.0)</end>
</program>
XML: 0.0200s,
[Total: 0.6810s]
-----
loac -visitor source hello.cs
[parse: 0.0109s]

class Hello {
    public static void Main() {
        System.Console.WriteLine("Hello world")
    }
}

[source: 0.0300s]
[Total: 0.6810s]
* users dir sharp compiler
  < process, [Process] End >

```

```

> .\bin\Debug\Browser.dll hello cs
/loac -visitor Browser BrowserVisitor debug/Browser.dll hello cs
/loac -visitor kind -visitor typecheck -visitor Browser BrowserVisitor debu
/loac -visitor typeswitch source demos typeswitch sample cs
* uses dsh sharp compiler inside sh .Shell File, CS*
  < using_directives>
    < sym / >
    < end-hello cs(0 2)< end>
  < /compilation_unit>
< /inputs>
  < assemblyRefs />
< /compilation>
< end>(0.0)< /end>
< /program>
[XML: 0.0200s]
[Total: 0.6810s]
-----
info -visitor source hello.cs
[parse: 0.6109s]

class Hello {
    public static void Main() {
        System.Console.WriteLine("Hello world")
    }
}
[source: 0.0300s]
[Total: 0.6810s]
-----
[process] [Process] End *
```

```

> .load -visitor Browser BrowserVisitor debug Browser.dll hello.cs
> .load -visitor kind -visitor typecheck -visitor Browser BrowserVisitor debug
> .load -visitor typeswitch source demos typeswitch sample.cs
x users drh csharp compiler inside sh .Shell file, cs
    if (yy_1 is int32) {
        int32 x = (int32)yy_1;
        Console.WriteLine(x);
        goto yy_1_end;
    }
    if (yy_1 is Symbol) {
        Symbol s = (Symbol)yy_1;
        Symbols.Add(s);
        goto yy_1_end;
    }
    if (yy_1 is Segment) {
        popSegment();
        goto yy_1_end;
    }
    throw new ArgumentException();
yy_1_end
}

[typeswitch_source 0 0501s]
[Total: 1.33219s]

(process, 'Process' End *)

```

The Easy Way Source-to-Source Transformation

- Prepare `mmio_struct_t` = `res_demo`
 `mmio_struct_t mmio_struct;`
 `mmio_struct = mmio_struct_init(&res_demo);`
 `mmio_struct = mmio_struct_init(&res_demo);`
- Alternative
 - `mmio_struct_t mmio_struct;`
 - `mmio_struct = mmio_struct_init(&res_demo);`

The Full 9 Yards Emitting MSIL

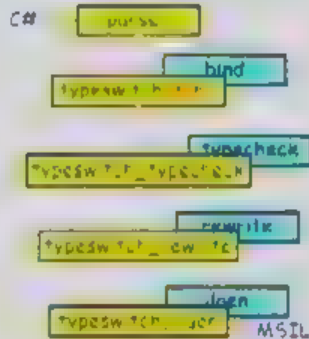
• Extend `entire.cs` to:

```
    emit
```

• `Assembly`

- `Module`

-



• • • • •

```
./load -visitor Browser BrowserVisitor debug Browser.dll hello.cs
./load -visitor Bind -visitor typecheck -visitor Browser BrowserVisitor debug
./load -visitor typeswitch source demos typeswitch sample.cs
x \users\dm\sharp\compiler\inside\sh\Shell\file1.cs
```

```
static Array<Type> symbols;
public static void Foo(object o) {
    object yy_1 = o;
    if (yy_1 is Int32) {
        Int32 x = (Int32)yy_1;
        Console.WriteLine(x);
        goto yy_1_end;

        if (yy_1 is Symbol) {
            Symbol s = (Symbol)yy_1;
            symbols.Add(s);
            goto yy_1_end;
        }
        if (yy_1 is Segment) {
            popSegment();
            goto yy_1_end;
        }
        throw new ArgumentException();
    }
    yy_1_end: ;
}
```

```
(process) [Process] 99% *
```

```
> . . . T>
/!end -visitor Browser BrowserVisitor debug Browser dll hello cs
/!end -visitor kind -visitor typecheck -visitor Browser BrowserVisitor debu
!end -visitor typeswitch source demos typeswitch-sample.cs
* users dh csharp compiler inside sh .Shell file, cs*
call instance void [mscorlib]System.Object::ctor()
ret
maxstack 1
} // end of method Test Test
method hidebysig specialname itspecialname static private void cctor()
ret
maxstack 0
, // end of method Test Test
} // end of class Test
}

* users dh csharp compiler typeswitch-sample.cs Fundamental) End
```

```

> 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
/ load -visitor Browser BrowserVisitor debug Browser dll hello cs
/ load -visitor kind -visitor typecheck -visitor Browser BrowserVisitor debug
load -visitor typeswitch source demos typeswitch-sample.cs
x users dir csharp compiler inside sh (Shell File, 65)
// demos/typeswitch-sample.cs:11:42: break
br $3
$5:
// demos/typeswitch-sample.cs:12:3) case Symbol (s)
ldloc 0 // t0
inst class Test/Symbol
locals init ((2)class Test/Symbol t2 )
stloc 2 // t2
ldloc 2 // t2
brfalse $7
ldloc 2 // t2
stloc 0 // s
// demos/typeswitch-sample.cs:12:20: Symbols.Add(s)
ldfld class [mscorlib,System.Collections.ArrayList Test Symbols
ldloc 0 // s
callvirt instance int32 [mscorlib,System.Collections.ArrayList Add (object
pop
// demos/typeswitch-sample.cs:12:16: break
br $3
$7:
// demos/typeswitch-sample.cs:13:3) case segment
ldloc 0 // t0
x users dir csharp compiler typeswitch-sample -- 'Fundamental') 62+

```



```

//lsc -visitor:Browser.BrowserVisitor debug/Browser.dll hello.cs
//lsc -visitor:bind -visitor:typcheck -visitor:Browser.BrowserVisitor, debu
//lsc -visitor:typeswitch_source demos/typeswitch-sample.cs
x:\users\drh\csharp\compiler\inside.sh [Shell Fill] 66*
.maxstack 0
} // end of method Test.popegment
.field static private class [mscorlib]System.Collections.ArrayList 'Symbols'
.method hidebysig public static void 'Foo'(object 'o') {
// demos/typeswitch-sample.cs(10,3): typeswitch (o)
.locals init ([0]object 't0')
ldarg 0 // o
stloc 0 // t0
// demos/typeswitch-sample.cs(11,3): case int32 (x):
ldloc 0 // t0
inst int32
.locals init ([1]int32 't1')
stloc 1 // t1
ldloc 1 // t1
brfalse $5
ldloc 1 // t1
stloc 0 // x
// demos/typeswitch-sample.cs(11,20): Console.WriteLine(x)
ldloc 0 // x
call void [mscorlib]System.Console::'WriteLine'(int32)
// demos/typeswitch-sample.cs(11,42): break:
br $3
x:\users\drh\csharp\compiler\typeswitch-sample.1 [Fundamental] 45*

```

```

./lsc -visitor:Browser.BrowserVisitor.debug/Browser.dll hello.cs
./lsc -visitor:bind -visitor:typeof -visitor:Browser.BrowserVisitor.debug
./lsc -visitor:typeof_source demos/typeswitch-sample.cs
x:\users\drh\csharp\compiler\Inside.sh [Shell Fill] 66
maxstack 0
// end of method Test.popsegment
.field static private class [mscorlib]System.Collections.ArrayList `Symbols'
.method hidebysig public static void `Foo'(object `o') {
// demos/typeswitch-sample.cs(10,3): typeof(o)
.locals init ([0]object `t
ldarg 0 // o
stloc 0 // t0
// demos/typeswitch-sample
ldloc 0 // t0
ldint int32
.locals init ([1]int32 `t1
stloc 1 // t1
ldloc 1 // t1
brfalse $5
ldloc 1 // t1
stloc 0 // x
// demos/typeswitch-sample.cs(11,20): Console.WriteLine(x)
ldloc 0 // x
call void [mscorlib]System.Console::`WriteLine'(int32)
// demos/typeswitch-sample.cs(11,42): break
br $3
x:\users\drh\csharp\compiler\typeswitch-sample.11 [Fundamental] 44

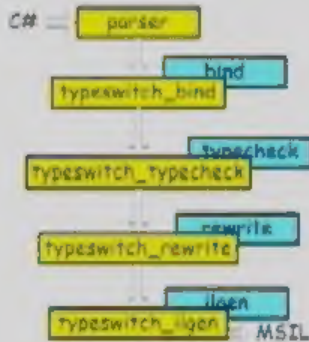
```



PowerPoint Slide Show - [Inside.ppt]

The Full 9 Yards: Emitting MSIL

-
-
- Extend bind.cs (100 lines)
- Extend typecheck.cs (35 lines)
- Extend rewrite.cs (29 lines)
- Extend ilgen.cs (65 lines)
- (Demo)
- Alternative
 - Build appropriate if-else ASTs directly, on rewrite AST as first visitor
 - Use bind.cs, typecheck.cs, rewrite.cs, ilgen.cs unmodified



Recap

- A simple C# compiler
 - ◆ Easy to understand (at least relatively)
 - ◆ Easy to modify
 - ◆ Ideal for language design experiments, language-based tools
- More info
 - ◆ <http://marweb.pla/csharp/compiler/>
 - ◆ Source Depot port MSR 5010, [//pla/csharp/compiler/](http://pla/csharp/compiler/)
 - ◆ Current snapshot in [\\research\Root\Public\drh\csharp\compiler](http://research.Root.Public.drh/csharp/compiler)